Proceedings for the 2015 PV Conference,
3-5 November 2015, Darmstadt, Germany

THE PDS4 INFORMATION MODEL DESIGN PRINCIPLES - HOW WELL DID THEY WORK?

John S Hughes[1], Daniel Crichton[1], Richard Simpson[2], Mitchell Gordon[3], Ronald Joyner[1], Anne Raugh[4], Edward Guinness[5], Michael Martin[6]

[1]Jet Propulsion Laboratory, California Institute of Technology, Pasadena, CA 91109, USA
[2]Stanford University, Stanford, CA, USA
[3]SETI Institute, Mountain View, CA, USA
[4]University of Maryland, College Park, MD, USA
[5]Washington University, St Louis, MO, USA
[6]ADNET Inc., Rockville, MD, USA

Abstract

The Planetary Data System (PDS) recently released Version 1.5 of the PDS4 Information Model, the primary component of the PDS4 Information Architecture. The Information Model is now stable and is in use by three active missions and several missions in various phases of development. The Information Model drives the PDS4 Information System using a multi-level governance structure that provides for common, discipline, and mission level management of the system's information standards. For the development of the PDS4 Information Model several design principles were adopted. This paper will describe each design principle and then explain how well they worked, problems encountered during development and their solutions, and how well the results meet the requirements of the multi-discipline planetary science community.

INTRODUCTION

The Planetary Data System (PDS), NASA's planetary science data archive is tasked to ensure the long-term preservation and usability, as well as near-term discoverability and distribution, of scientific data returned by all NASA supported missions to explore the solar system, except for those directly related to the study of the Sun or the Earth. The resulting archive is large, both in terms of data volume and complexity. Planetary science targets include the major planets, dwarf planets, comets, asteroids, Kuiper Belt Objects, satellites, rings, dust, and the fields and charged and neutral particles which pervade the entire solar system. Some data is obtained remotely while some is obtained through *in situ* measurements. The current archive contains almost one petabyte of data obtained by more than 1,257 unique instruments. In order to accomplish these tasks, the PDS system is distributed and managed by science domain experts at separate physical locations known as nodes. [1, 6]

After two decades of operation, the PDS embarked on a complete system redesign starting with the definition of an explicit, over-arching system architecture that would scale to meet the demands of higher data volume and leverage new information technologies. This next generation system is called PDS4.

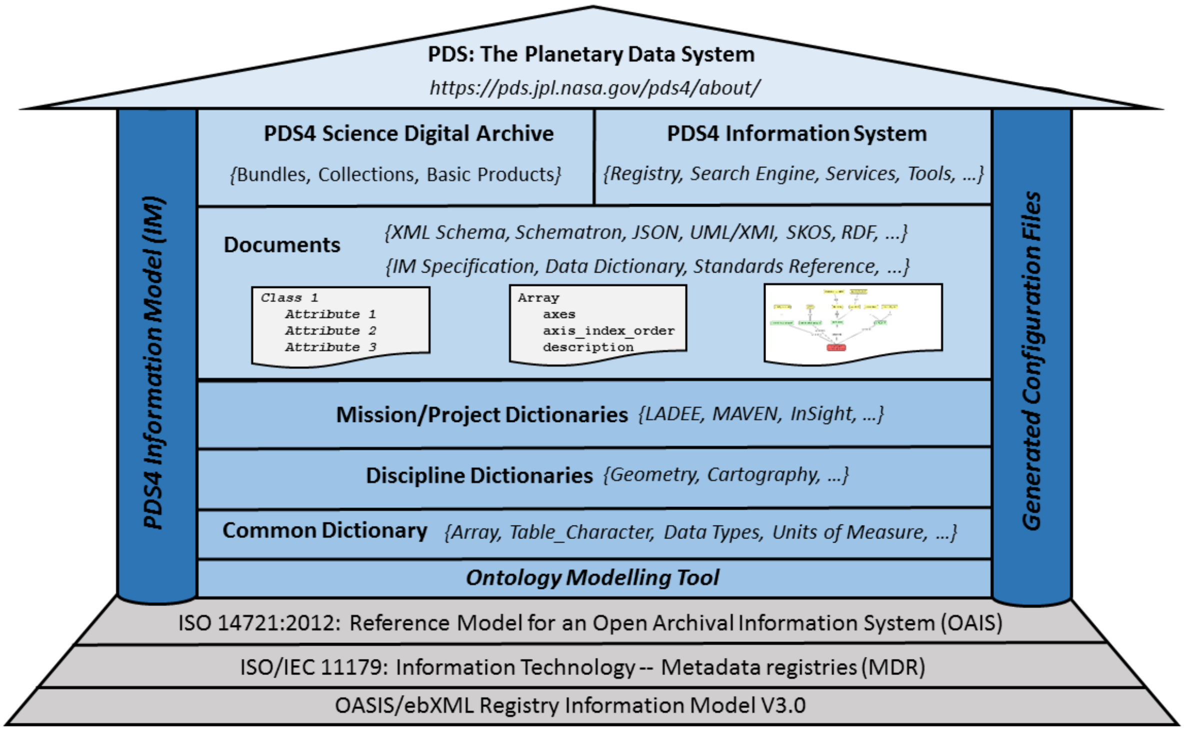
The PDS4 architecture has two primary components: the information architecture and the software/technical architecture. The PDS4 Information Architecture (IA) [3, 4, 5] allows all PDS data to be described using a common model based on the ISO 14721 reference model [7]. It uses the Extensible Markup Language (XML) [9], a widely accepted and well-supported standard for data product labelling, validation, and searching. The IA supports a hierarchy of data dictionaries built to the ISO/IEC 11179 standard [8] and designed to increase flexibility, enable complex searches at the product level, and to promote interoperability that facilitates data sharing nationally and internationally.

PDS4 provides a hierarchical structure for data archiving with three types of products. The *Bundle Product* is a list of all related collections. The *Collection Product* is a list of related basic products of similar type (all raw images from a single instrument or all documents from a mission, e.g.). The *Basic Product* is the smallest unit of data registered and tracked in the PDS (a single image, table, or document). The model defines four fundamental data structures: *Array* - a homogeneous n-dimensional array of scalars (e.g., images or spectral qubes); *Table* – the traditional fixed-width structure based on a single record with heterogeneous binary or character fields; *Parsable Byte Stream* – a stream from which the data value can be extracted directly by applying parsing rules to the bytes (e.g., simple text files, XML files, CSV tables); *Encoded Byte Stream* – a stream in which the bytes must interpreted, transformed, or otherwise processed before the data can be extracted (e.g., PDF files, JPEG images, MPEG movies).

The Software/technical Architecture (SA) is a distributed service-oriented architecture encompassing the individual PDS discipline nodes and the PDS's international partners. The SA provides consistent protocols for access to the data and services and an open source registry infrastructure to track and manage every product in the PDS. Finally the SA supports a distributed search infrastructure in which product metadata is extracted from the registry and loaded into Apache Solr.

PDS4 is the first operational science information system resulting from an information model-driven development methodology [2]. It is being used to coordinate data archiving in both the national and international planetary science communities. With the system's information requirements captured in an ontology modelling tool significant but controlled change can occur as the science domains and implementation technologies change.

The following section outlines the design principles for the Information Model, the core component of the PDS4 Information Architecture. These principles will be explained and their implementation evaluated. The components of the PDS4 architecture are illustrated in Figure 1.



PDS: The Planetary Data System
https://pds.jpl.nasa.gov/pds4/about/
PDS4 Science Digital Archive
{Bundles, Collections, Basic Products}
PDS4 Information System
{Registry, Search Engine, Services, Tools, ...}
Documents
{XML Schema, Schematron, JSON, UML/XMI, SKOS, RDF, ...}
{IM Specification, Data Dictionary, Standards Reference, ...}
Class 1
Attribute 1
Attribute 2
Attribute 3
Array
axes
axis_index_order
description
Mission/Project Dictionaries {LADEE, MAVEN, InSight, ...}
Discipline Dictionaries {Geometry, Cartography, ...}
Common Dictionary {Array, Table_Character, Data Types, Units of Measure, ...}
Ontology Modelling Tool
PDS4 Information Model (IM)
Generated Configuration Files
ISO 14721:2012: Reference Model for an Open Archival Information System (OAIS)
ISO/IEC 11179: Information Technology -- Metadata registries (MDR)
OASIS/ebXML Registry Information Model V3.0

Figure 1 - PDS4 Systems Architecture

DESIGN PRINCIPLES

The PDS4 Information Model is the cornerstone of the model-driven PDS4 Information System in that the Information Model drives both the development and the maintenance of the information system. The success of the model-driven approach is dependent on a few key design principles that were identified during the architectural design phase of PDS4 development. These include the identification and use of a standard information systems architecture, the insistence on attainable goals within a well-defined scope, the design of an independent Information Model, the use of knowledge acquisition to obtain input from domain experts, and the development of a multi-level governance structure.

Standard Information Systems Architecture

A key principle in the development of PDS4 was the use of a standard information systems architecture. [17] Three components were identified — the information, technology, and process architectures. The information architecture in particular was defined as being independent and thus became the driver for the technology and process architectures. This approach allowed parallel development of the three components with the requirement that the technology and process architecture respond to the information architecture during both development and operations. This also allows a more controlled and less disruptive evolution of the system since the information architecture evolves with the science disciplines while the technology architecture evolves with the computer industry.

Well-Defined Scope and Goals

With 25+ years of institutional experience, and the priincipal designers all having at least 10 and up to 20 years of personal experience archiving and supporting planetary science data, the PDS4 development team had a good understanding of NASA's planetary science community and the capabilities that were being requested, for example easy search and retrieval, stable data formats and ease of use, and long-term preservation. From this experience base, the team developed a well-defined scope in the form of a multi-level set of requirements to meet the current and more advanced expectations of the community. Additionally, a detailed analysis of the evolution and content of the current archive, containing more than 30 years of solar system exploration data, provided valuable insight for defining boundaries for data and formats. Several key goals for the development of PDS4 could then be identified.

Prior experience established that accepting data in "any format" and requiring thorough description of the data structure did not necessarily make the data easy to use. Consequently, a primary goal was limiting the data formats in the archive to a few simple fundamental data structures that would remain stable and usable over time.

In general the goal of the Information Model was to define the "things of interest" in the planetary science community to a level where both the functional and long-term requirements of the archive could be met. "Things of interest" included the objects to be processed (for example, digital images and time-series), the descriptions that provided context (hardware and calibration descriptions), and the relationships among things that provided meaning (the links from an image to a document containing the mission science objectives and to another document describing the camera system).

Implementation Independent

The independence principle required that the Information Model remain independent of any implementation, including the choice of the implementation language for the Information Model itself. This principle was adopted to address issues that might arise due to technology evolution. For example the storage technology used by the PDS has evolved from tape, to CD-ROM, to rotating disk storage as the total volume of the data in the archive has increased from a few gigabytes to currently almost one petabyte. The data management technology likewise evolved from relational databases, to text- and facet-based search engines. Data definition languages evolved from the home-grown Object Description Language (ODL) [14] to XML and its extensions and Resource Description Framework (RDF) [12] and its variants. And of course the Internet allowed solitary data repositories to transform into networked online resources.

More specifically, the primary benefit of an independent Information Model is that it allows the model to evolve at a different speed from the chosen implementation technology, thereby disentangling the model from that technology. Prior to PDS4, the PDS data model was tightly bound to the ODL definition. In many cases new data types had to be jury-rigged into the system resulting in inconsistencies and ambiguities.

To address the independence principle for the PDS4, the ontology modelling tool *Protégé* [15] was adopted to capture the Information Model in the most expressive language available. Since the modelling language is semantically richer than most standard data definition languages, subsets of the Information Model could be extracted for specific system and user needs.

In science domains, especially those that are international, many attributes require extensions — for example, for units of measure, patterns of non-decimal values, and names and definitions in other natural languages. The ISO/IEC 11179 metadata registry reference model was adopted to capture this information and this augmentation resulted in a companion ontology – the metamodel used to define the PDS4 Information Model.

Knowledge Acquisition from Science Domain Experts

The single most difficult challenge in the development of the PDS4 Information Model was the capture of knowledge from planetary science domain experts. The Data Design Working Group (DDWG) was formed with at least one participant from each of the PDS nodes. On average there have been about 20 individuals actively involved at any time. Most of the individuals had significant PDS experience. The team agreed to start by designing from first principles, the first being “a few simple data structures”, and calling on their collective experience in all relevant design decisions.

The ontology modelling tool allowed each item to be formally defined as it was being designed. Documents and graphics were generated to allow designers to review and comment as soon as possible. The designs were reviewed with respect to suitability and the level of detail needed to meet archive requirements.

The large number of categories of information required for an archive increased the difficulty of the task. These included digital object structures (data formats), context, representation, integrity, provenance, and reference information, and containers such as Products, Collections, and Bundles. Also the domain experts critical for this task required training in data modelling (for example, data normalization and object-oriented modelling).

Multi-level governance

Prior to PDS4, the PDS data model was monolithic with one governance entity — essentially the entire PDS. Gaining consensus among a large group of experts often is difficult. This is particularly true for planetary science data archives where some parameters are essential for some disciplines and completely irrelevant for others. Consequently, under PDS3, making changes often resulted in long delays and frustration both inside the organization, and with external organizations preparing new material for submission.

The PDS4 multi-level governance structure provides for common, discipline, and mission level management of the model. The model is partitioned into namespaces, each under the control of a steward. All stewards are under a single registration authority.

Mission Drivers

The development of an Information Model for a diverse community such as planetary science is a complex and time consuming task, as evidenced by the several years of work required to design and release the first version of the PDS4 Information Model. However as the model matures there is a tendency to revisit and “clean up” prior designs. New insights also suggest better designs. The potential is never being quite ready for the first release.

After the model had matured, a mission was chosen that was willing to be a beta test subject – enabling PDS to do an end-to-end test of the new standards. Version 1.0 of the Information Model was released, placed under configuration control, and product labels were designed for the Lunar Atmosphere and Dust Environment Explorer (LADEE) mission.

LESSONS LEARNED

Standard Information Systems Architecture

Early definition of a PDS4 system architecture consisting of three independent components — the information, technology, and process architectures — was critical to the success of PDS4. Even though development of the information architecture took over four years, when it was first released for use the system components were integrated and deployed in a relatively short period of time. There was some impact on the System Design team in the early stages when the model was relatively volatile resulting in multiple instances of the design team modifying interfaces.

In the two years since the first deployment, the information, technology, and process architectures have continued to mature separately. For periodic builds a candidate Information Model is released, the build enters into the system integration and test phase, and the resulting system is deployed. The system integration and test phase now takes approximately one week to complete.

Well-Defined Scope and Goals

The well-defined scope and goals, with the resulting requirements, allowed development of PDS4 with no major changes of course. In particular the experiences and lessons learned over 20 years helped to define stable architectural components while also ensuring flexibility. In general the primary goals were always in view. This allowed minor course corrections when necessary.

Deployment of the Information Model meant that subsequent changes had to be managed. The Information Model was placed under configuration control and a Change Control Board (CCB) was established to review changes and decide whether the benefits of the proposed change justified the impact on the system and the community.

Implementation Independent

Capturing the Information Model in an ontology modelling tool was technically easy. The ontology tool worked as advertised. As the “things of interest” in the domain, their components, and relationships were identified, their definitions were entered into the modelling tool in fairly obvious and logical ways. The ontology tool was simple to use but enforced a disciplined and consistent modelling paradigm.

The adoption of the ISO/IEC 11179 reference model provided the attribute extensions required for PDS4. A wide range of issues were resolved from how to categorize attributes to capturing the meanings of permissible values. The implementation of the ISO/IEC 11179 reference model in the ontology model tool was relatively easy and required little customization.

The translation of the ontology contents to the chosen implementation language, XML Schema [10, 11], was also technically easy but required the careful selection of XML Schema constructs that carry the same meaning. For example, classes and their properties that were defined in the ontology were implemented in XML Schema as either `xs:simpleType` or `xs:complexType` depending on the complexity of the definition. However either type was defined as an XML Schema `xs:element` for use in XML documents.

Once the Information Model had been translated to XML Schema, an issue arose that the Information Model seemed to “exist” in two languages. Designers who had become comfortable with an “objected-oriented” model now encountered the same objects in XML Schema, but defined and organized in a significantly different way.

For some individuals the XML Schema version became the version of choice. This raised issues when XML constructs were proposed to extend the model. For example, to improve XML document processing efficiency the use of XML Path Language (XPath) was proposed for referencing. So the PDS4 oversight (DDWG, System group, and CCB) have to be diligent in making sure that any updates preserve the implementation-independence of the IM and ontologies.

The independence principle has worked as expected. The Information Model, the registry, and the system were developed on parallel paths. Once developed the system services and tools responded as expected to the Information Model via extracted configuration files. This approach has proven to be so effective that developers of other systems have requested that the Information Model be translated to JSON [19], SKOS [18], and UML/XMI [13] in order to configure pipelines and support Linked Open Data (LOD) applications. The task of writing a translator takes a day or two.

Knowledge Acquisition from Science Domain Experts

Development was multi-staged. The initial stage involved a small cadre that established the primary goals, defined the underlying data structures, and produced the skeleton for the model. The intermediate stage expanded the working group (the Data Design Working group, DDWG), developed the aggregation of products within the model and fleshed out the skeleton by concentrating on the basic pieces that needed to be in place prior to the first release. In the late stage, the focus of the DDWG shifted to addressing more detailed parts intentionally postponed until the basics were in place, resolving unforeseen issues, developing discipline and mission level models, and refining the model through support of the CCB. A significant aspect of the development was the small size (five members) of the working group during the initial phase. The diversity of backgrounds, archiving experience, and experience using data, ensured an initial design with the capability to support the broad needs of PDS and the communities it supports, while the small size of the initial cadre made consensus easier to reach.

Knowledge acquisition is difficult. The benefit is that the resulting Information Model provides a common domain of discourse - definitions of the domain “items of interest” - that allows effective communication between domain scientists, computers, and users.

Although consensus often becomes more difficult as the number of discussion participants increases, it is important to have thoughtful input from all stakeholders and to ensure a balance between science and IT experts so that benefits and costs can be properly weighed. Good note-taking is important. During PDS4 development, design decisions were quickly entered and tested in the modelling tool. However, the detailed reasons for a decision and supporting discussions were not always captured. Later, many decisions were revisited. Detailed information about how a previous decision had been made would have been beneficial. Decisions reached hastily were among the most likely to be revisited.

Multi-level governance

The partition of the PDS4 Information Model into namespaces managed by stewards has been effective. The PDS4 common model, even though governed by the entire PDS, is smaller and has quickly become table.

New design work is now focused on the development of the cross-cutting discipline models such as geometry and cartography. Smaller teams of discipline experts develop consensus faster than would be expected by having the entire DDWG involved. Finally Mission level models are developed by a few of those most intimately involved and the previous lesson regarding note-taking is being applied.

Mission Drivers

The decision to release V1.0 for use by the LADEE mission was a critical milestone in the development of the Information Model. In general the model performed well but some issues were identified. To manage change and maintain stability, subsequent releases were scheduled six months apart and a Change Control Board (CCB) was formed to review all changes. Approval of a change is based on whether the impact of the change on the community is warranted with respect to the expected benefits.

The use of the Information Model by the LADEE mission provided the testing required to mature the common model. Most of the changes now requested are either bug fixes or requests for new permissible values.

Local Data Dictionaries (LDDs) are designed at the discipline and mission level. To maintain consistency with the common model an XML template called Ingest_LDD has been designed to capture LDDs. A completed Ingest_LDD template is ingested into the master database to test for consistency against the common model and the modelling methodology. For example all references to data types and units of measure are validated against their definitions in the common model.

CONCLUSION

PDS4 has been and continues to be a good case study of a model-driven architecture. The Information Model drives the PDS4 Information System using a multi-level governance structure that provides for common, discipline, and mission level management of the system’s information standards. The development of this system followed several design principles. These principles were applied consistently throughout the task and have resulted in an operational system. The result is a stable common model with additional work continuing in the development of discipline, cross-cutting, and mission level models. Most of these additions have been routine extensions to the existing models.

ACKNOWLEDGEMENTS

The research was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.

The authors wish to acknowledge the PDS Data Modelling Team and the PDS Technical staff who designed, implemented, and kept the PDS data standards and systems infrastructure viable over two decades in the continually evolving planetary science domain. They would also like to acknowledge the PDS4 Data Design Working Group (DDWG), the Systems Design Working Group (SDWG), and the International Planetary Data Alliance (IPDA) for their significant efforts in the design, development, and implementation of the PDS4 information and system architectures. These discipline experts remained committed, sought excellence, and provided first-rate information without which PDS4 would not have been possible. Finally, they would like to recognize the support of the PDS Management Council and NASA Headquarters.

REFERENCES

[1] Arvidson, R., ed., Issues and Recommendations Associated with Distributed Computation and Data Management Systems for the Space Sciences, Committee on Data Management and Computation (CODMAC), National Academy Press, 1986.

[2] Crichton, D., Hughes, J.S., Hardman, S., Law, E., Beebe, R., Morgan, T., Grayzeck, E.*, A Scalable Planetary Science Information Architecture for Big Science Data*, 10th IEEE e-Science conference, 2014.

[3] Crichton, D., Beebe, R., Hughes, S., Stein, T., Grayzeck, E., “PDS4: Developing the Next Generation Planetary Data System”, EPSC Abstracts, Vol. 6, EPSC-DPS2011-1733, EPSC-DPS Joint Meeting 2011.

[4] Hughes , J.S., Crichton, D., Hardman, S., Law, E., Joyner, R., Ramirez, P.*, PDS4: A Model-Driven Planetary Science Data Architecture for Long-Term Preservation*, IEEE 30th International Conference on Data Engineering (ICDE), Chicago, IL USA, 2014.

[5] Hughes, J.S., Crichton, D. J., Mattmann, C. A., “Ontology-Based Information Model Development for Science Information Reuse and Integration”, 10.1109/IRI.2009.5211603, IEEE International Conference on Information Reuse & Integration, 2009.

[6] Special Issue: The Planetary Data System, Planetary and Space Science, European Geophysical Society, ISSN 0032-0633, Volume 44, Number 1, January, 1996.

[7] ISO 14721:2003: Reference Model for an Open Archival Information System (OAIS), ISO, 2003.

[8] ISO/IEC 11179: Information Technology -- Metadata registries (MDR), ISO/IEC, 2008.

[9] Extensible Markup Language (XML) 1.0 (Fifth Edition), W3C Recommendation, 26 November 2008.

[10] XML Schema Part 1: Structures Second Edition, W3C Recommendation, 28 October 2004.

[11] XML Schema Part 2: Datatypes Second Edition, W3C Recommendation, 28 October 2004.

[12] W3C RDF/XML Syntax Specification (Revised), W3C Recommendation, 10 February 2004.

[13] OMG MOF 2 XMI Mapping Specification, OMG, Version 2.4.1, June 2013.

[14] Planetary Data System Standards Reference, Chapter 12. Object Description Language Specification and Usage, Version 3.8, February 27, 2009.

[15] (2013) The Protégé Ontology Editor and Knowledge Acquisition System website. [Online]. Available: http://protege.stanford.edu/.

[17] Reference Architecture for Space Information Management (RASIM), CCSDS 312-0.G-1.

[18] SKOS Simple Knowledge Organization System Reference, W3C Recommendation, (2009).

[19] The JavaScript Object Notation (JSON) Data Interchange Format, Internet Engineering Task Force (IETF), 2014.